Middle East Technical University
Department of Electrical and Electronics Engineering
EE230: Probability and Random Variables
Term Project

Due: 23:59 on April 12, 2020

Photodetectors are devices that sense electromagnetic energy, typically light, and convert it into an electrical signal. They are used in various technologies in imaging, remote sensing, and communications. Examples include devices that automatically open supermarket doors, sensors on TV remote controls, CCD image detectors in cameras, and huge telescopes used by astronomers to detect and study the radiation from our amazing universe [1].

Quantum theory of light[2] tells us that the light is made of particles (photons), which are emitted by the source at random. As a result, the amount of photons emitted by a source (sun, laser, bulb, etc.) and then hit a photodetector is not constant, but subject to random fluctuations.

In this project, we will first investigate how to model the randomness of photons using Poisson distribution. Then we will explore how this randomness will come into the picture as *noise* [3] in the applications of photodetectors. Understanding this random phenomena will in general be important for the design and analysis of optical imaging and communication systems.

PART 1: MODELING

A photodetector effectively counts the number of photons incident on the sensor (pixel) over a given time interval. We start by modeling this number of photons based on the following commonly held assumptions:

- When Δ is sufficiently small, the probability of one photon arriving in a time interval Δ is proportional to Δ. That is,

$$P(\text{One photon arrives on the sensor during a time interval } \Delta) = a\,\Delta \tag{1}$$

  where a is a constant that is proportional to the size of the sensor and the brightness of the scene, and satisfies $a\,\Delta < 1$. The unit for Δ will be taken as second (sec) (hence the unit of a is 1/sec).

- When Δ is sufficiently small, the probability that more than one photon arrives in the interval Δ is negligible:

$$P(\text{More than one photon arrives during the time interval } \Delta) \approx 0 \tag{2}$$

- The number of photons that arrive in one time interval is independent of the number of photons that arrive in any other non-overlapping interval.

[1] More information about photodetectors and their applications can be found in the following link: `https://www.laserfocusworld.com/detectors-imaging/article/16555513/introduction-to-photodetectors-and-applications`

[2] More information about quantum theory of light can be found in the following link: `http://www.grandinetti.org/quantum-theory-light`

[3] More information about this type of noise can be found in the following link: `https://en.wikipedia.org/wiki/Shot_noise`

Q1)

a) Let X be the number of photons arriving on the sensor in a short time interval of length Δ, with Δ satisfying all the three assumptions above. Clearly write the PMF of X, and express its expected value in terms of a and Δ.

b) Now write a Matlab code to generate this random variable for given a and Δ. (Your code should output one of the possible values of X, which is generated randomly based on the PMF of X. Here you need to use a "Monte Carlo Method" similar to the one in your first homework. You will again choose a random number uniformly distributed between 0 and 1, and then use this now to decide whether a photon has arrived.) Set $a = 10$ and $\Delta = 0.005$. Run your code 10^5 times to generate 10^5 realizations of X values and draw the histogram of these values. Normalize the vertical axis of the histogram with 10^5. (For this, you may use the "normalization" property of the built-in Matlab function "histogram"). Does this histogram look similar to the PMF of X? The answer should be yes based on the theory we will learn in Chapter 5 (see, for example, Example 5.4 in the book). If your answer is no, then your code must be wrong, and you need to correct it before you proceed to the next Matlab task.

Q2)

a) Suppose now the recording (integration) time of the sensor is t. (In a camera, t is called the exposure time, and is controlled by the camera shutter.) This recording time t can be composed into n time intervals, each with length Δ, that is we have $t = n\Delta$. Assume that for each of these time intervals the constant a is the same. Define independent Bernoulli trials with each trial counting the number of photons in one of the time intervals. Set $a = 10$, $\Delta = 0.005$, and $t = 1$, and write a code to generate these Bernoulli trials. Your code should output a vector of ones and zeros. Plot your result. In this plot, y-axis should be the number of photons arrived in each time interval, and x-axis should contain values from 1 to n. Rerun your code for a total of 10 times, and compare the resulting plots. Do you think there is anything common in these plots? Comment.

b) Now let Y be the total number of photons recorded on a sensor during the exposure time t (as in part a, $t = n\Delta$). Based on part a, you can consider Y as sum of independent Bernoulli random variables. What type of random variable is Y? (Bernoulli, Binomial, Geometric, Poisson,...) Also write the PMF of Y by clearly expressing its parameters. Also find its expected value in terms of only a and t. What does this expected value mean in terms of the photons hitting this sensor?

c) Set $a = 10$, $\Delta = 0.005$, and $t = 1$. Modify your code in part a to generate the random variable Y. Then use it to generate 10^5 realizations of Y and plot their histogram. Normalize the vertical axis of the histogram with 10^5. To compare this histogram to the PMF of Y, also plot the known PMF of Y with correct parameters. Verify that the normalized histogram look similar to the PMF of Y.

Q3)

a) Remember that the assumptions made in modeling require Δ to be sufficiently small. Now we want to investigate the distribution of Y as Δ goes to zero (i.e., n goes to infinity), while keeping the exposure time $t = n\Delta$ constant. What does the PMF of Y converge to in this case? Compute and clearly write the limiting PMF. At this point you should be able to identify that Y can be approximated as Poisson distributed. Specify the parameter of this Poisson distribution in terms of a and t. Also specify its expectation.

b) Now plot the PMF for the distribution found in Q2 part b. Generate the plot for the following cases:

- $a = 10$, $t = 1$ second, and $\Delta = 0.05$ second
- $a = 10$, $t = 1$ second, and $\Delta = 0.005$ second
- $a = 10$, $t = 1$ second, and $\Delta = 0.0005$ second

Also modify your code to plot the PMF for the Poisson distribution obtained in part a. Use $a = 10$ and $t = 1$ second. Compare these plots and comment.

c) Find $P(Y < 2)$ by using the found PMF in Q2 and also the Poisson distribution in Q3. Express these two probabilities explicitly in terms of a, t, and Δ. Now evaluate them for the parameter values given in part b. Compare the results and comment on their agreement.

PART 2: APPLICATIONS

In the modeling part of the project, we have investigated how to model the randomness of photons using Poisson distribution. In particular, we have showed that the number of photons arriving to a photodetector during its integration time is Poisson distributed. Now we will explore how this randomness comes into the picture as *noise* in the applications of photodetectors. In particular, we will use this Poisson model to explore the variability of the photon counts and how that affects the performance of optical imaging systems (such as our eyes or traditional cameras).

Q4) (Effect of shot noise on image quality)
Suppose you were wandering in one of the beautiful forests in Australia, and saw a cute koala sleeping on a tree. As an enthusiastic tourist, you wanted to take a picture of the koala with your camera.

a) Let us first consider that it was noon. In this case, suppose the average number of photons hitting each pixel on the camera over the recording (exposure) time is as given in the "koala.mat" file. The below code shows how to read this file, extract the average number of photons in a matrix named 'I', and plot in Matlab:

load('koala.mat','I')
imshow(I,[min(I(:)) max(I(:))])

Here the matrix I contains the average number of photons hitting each pixel on the CCD detector within a fixed exposure time. The size of the detector is 250×250 pixels. That is,

$I_{i,j}$ is the average number of photons hitting pixel (i, j) where $i, j = 1, 2, ..., 250$. If there were no random fluctuations in the number of photons arriving, you would obtain an image of the koala as it was plotted above (with the imshow() function) regardless of the lighting conditions. However, as derived in the modeling part, photons arriving to each pixel is Poisson distributed. That is, there are random fluctuations on the number of photons hitting each pixel, and hence on the image obtained with the camera. Here we are assuming that each detector pixel counts the number of photons and displays this count. Moreover, photon counts at each pixel are jointly independent. Using Matlab, the matrix I, and the Poisson model, generate two such possible image data of size 250×250 and plot each using the imshow() command. These correspond to the possible images of the koala that can be taken.

Hint: Note that the number of photons collected at each pixel is Poisson distributed, and you can generate realizations of Poisson random variables using the random() command in Matlab. From the Help menu of Matlab window, select Product Help, and in the new window that opens, type 'random' to search and read the documentation of the random() command to understand how to generate realizations from Poisson distribution. Moreover, to plot the photon counts and hence the image, you can use the imshow() command in the same way it was used above.

b) Now we will investigate how the image of the koala would look like if you took its picture at different light levels, for example close to sunset. Remember from the first part of the project that the expected number of photons arriving is proportional to the brightness of the scene; so as the sunset approaches, the expected number of photons arriving to camera has to decrease. To investigate the effect of this on the quality of the camera image (when all the camera settings are kept the same), scale the values in the 'I' matrix with 1/10, 1/100, and 1/1000, and repeat part a) to generate multiple realizations of the camera image (i.e. Poisson process) for each case. Compare the resulting images, and comment on how lower levels of light affects the quality of the image obtained. In particular, clearly explain why you observe a grainy structure at lower levels of light, and try to relate this to the ratio of the mean of the photon counts (for each pixel) to its standard deviation. (Hint: The source of this grainy structure is known as *shot noise* and it is the dominant source of noise in applications involving photodetectors and low-light levels.)

c) To prevent obtaining such noisy images at low-light levels, today's cameras resort two common things: 1) using flash, or 2) increasing the exposure time when the flash is off. Explain how these two approaches help to deal with the shot noise. Also try to comment on what other things you could possibly change on the camera to better deal with shot noise (such as changing the pixel size, camera aperture size,...), and why these changes can be undesirable.

Q5) (Estimating the brightness of a star)

In this question, we will illustrate how astronomers use probability theory to infer the brightness of a star. Imagine that an astronomer points a photon counter (such as a telescope) to the sky, and observes the light coming from a remote star. Suppose a series of N measurements is obtained with the telescope, and each measurement records the number of photons arriving at the telescope in one minute. (Here we will assume that the brightness of the star is constant during these N observations.) Let x_i denote the ith measurement (which has Poisson distribution), and λ denote the expected number of photons arriving at the telescope per minute. By knowing the telescope parameters and the λ parameter, the astronomer can infer the brightness of the star. But how will s/he estimate λ given the set of independent measurements $\{x_1, ..., x_N\}$? One approach that s/he can use is *maximum likelihood estimation*. Here we will investigate this approach.

a) Let us first derive a closed-form expression for the maximum likelihood estimate of λ. To learn about this estimation approach, you should read pages 462-465 from your book (in Section 9.1). In short, to find the *maximum likelihood estimate* of λ, you first need to find the joint PMF of the independently taken observations $\{X_1, X_2, ..., X_N\}$. Let us denote this with $p_{X_1,X_2,...,X_N}(x_1, x_2, ..., x_N; \lambda)$ (note that λ is an unknown parameter, and $\{x_1, ..., x_N\}$ are observed quantities). The maximum likelihood estimate will then be the value of the λ parameter that maximizes the function $p_{X_1,X_2,...,X_N}(x_1, x_2, ..., x_N; \lambda)$. That is, we choose the λ parameter that makes the observed data "most likely", or in other words, maximizes the probability of obtaining the data at hand. Following this approach, derive a closed-form expression for the maximum likelihood estimate of λ in terms of the observations $\{x_1, ..., x_N\}$.

Hint: For an easier computation, consider first taking the logarithm of the likelihood function $p_{X_1,X_2,...,X_N}(x_1, x_2, ..., x_N; \lambda)$ and then take derivative with respect to λ to find the optimal value. If you use this approach, explain why taking the logarithm does not change the optimal value of λ.

b) Let us now illustrate the maximum likelihood estimation with a numerical example. Set $N = 100$, and the unknown parameter $\lambda = 100$. Use Matlab to generate a realization of $X_1, X_2, ..., X_N$ (or equivalently, 100 realizations of a Poisson random variable). Suppose these are the observations obtained with the telescope. Using your answer to part a, find the maximum likelihood (ML) estimate of λ, given these observations. Comment on the resulting estimation error. Also for the given observations, plot the function $p_{X_1,X_2,...,X_N}(x_1, x_2, ..., x_N; \lambda)$ as a function of λ (using Matlab). Verify that the value of λ corresponding to the maximum value of this function is your maximum likelihood estimate.

NOTE: Matlab code written for each task must be included in a readable format to the end of your solution file. Otherwise you will get zero credit from that part.